

Aula 53 – Projeto Final – Parte 3: Planejamento Ágil

Mãos à Obra no Mundo Ágil: Construindo o Backlog e Planejando sua Primeira Sprint

Imagine a cena: você, como gerente de um novo projeto de TI, recebe um desafio. A diretoria quer um aplicativo inovador para estudantes universitários, mas a ideia ainda é um rascunho. Eles querem ver resultados rápidos, algo tangível em poucas semanas, não em meses. O método tradicional, com seu planejamento extenso e detalhado, parece lento demais para essa realidade. Como transformar uma visão ampla em ações concretas, entregando valor de forma rápida e contínua? É exatamente essa a habilidade que você começará a dominar hoje.

Esta aula é a ponte entre a teoria e a prática. Ao final destes 75 minutos, você não vai apenas entender o que é um *Product Backlog* ou uma *Sprint*; você será capaz de traduzir uma grande ideia em pequenas entregas de valor, as famosas **Histórias de Usuário**. Mais importante, você aprenderá a organizar e priorizar o trabalho para a primeira fase de execução do nosso projeto, a *Sprint Planning*, garantindo que sua equipe comece com o pé direito, focada no que realmente importa para o cliente.

Nossa jornada nos levará a criar a espinha dorsal do nosso projeto: a grande lista de desejos, conhecida como *Product Backlog*. Em seguida, aprenderemos a arte de escrever Histórias de Usuário que capturam a essência das necessidades do cliente. Por fim, simularemos o ritual que dá início a toda a ação no Scrum: a reunião de planejamento da *Sprint*. Este é o momento em que a estratégia se transforma em um plano de ação executável, uma habilidade essencial no mercado de trabalho atual, que valoriza a agilidade e a capacidade de adaptação.

O Coração do Projeto Ágil: O Product Backlog

Você já tentou organizar uma grande viagem em grupo? No início, todos têm ideias e desejos: "Vamos para a praia!", "Quero visitar museus", "Precisamos de um hotel com piscina", "E o voo, já viram?". Se você simplesmente anotasse tudo em um papel sem ordem, o caos seria inevitável. Como decidir o que reservar primeiro? O que é essencial e o que é apenas um "seria legal se"? Essa confusão é o ponto de partida de muitos projetos, um problema que o planejamento ágil resolve com uma ferramenta simples, mas poderosa.

❏ O **Product Backlog** é a solução para esse caos. Pense nele como a grande lista de desejos da sua viagem. É uma fonte única e ordenada de tudo o que é necessário para o projeto.

Não é um documento estático, trancado em um cofre; é um artefato vivo, que respira e evolve. Assim como na sua viagem, novas ideias podem surgir ("Que tal um passeio de barco?"), e prioridades podem mudar ("A previsão do tempo é de chuva, talvez devêssemos focar nos museus primeiro"). O *Product Owner*, o "dono do produto", atua como o organizador principal dessa viagem, garantindo que os desejos mais importantes estejam sempre no topo da lista.

Em nosso projeto de aplicativo para estudantes, o *Product Backlog* começaria com itens amplos como "Cadastro e Login de Usuário", "Busca de Atividades Complementares", "Sistema de Inscrição" e "Emissão de Certificados". Cada um desses itens representa uma entrega de valor para o usuário final. A beleza dessa abordagem é que não precisamos detalhar *todos* eles agora. Detalhamos apenas os que estão no topo, aqueles que planejamos fazer em breve. Isso reflete uma tendência crucial de 2025 na gestão: a **gestão híbrida**, onde temos uma visão geral do projeto (quase como um plano tradicional), mas detalhamos e executamos em ciclos curtos e adaptáveis, no estilo ágil.

Isso nos leva a uma questão fundamental: como transformar esses itens amplos em algo que a equipe de desenvolvimento possa realmente entender e construir?

A Linguagem do Valor: Escrevendo Histórias de Usuário

Dizer a uma equipe de desenvolvimento "Precisamos de um login" é como dizer a um chef "Prepare comida". É vago. Que tipo de comida? Para quem? Para qual refeição? A falta de clareza gera retrabalho, frustração e, o pior de tudo, um produto que ninguém quer usar. O verdadeiro desafio não é construir a funcionalidade, mas construir a funcionalidade *certa*, da maneira *certa*, para a pessoa *certa*. Como podemos garantir essa clareza?

A solução está em mudar a perspectiva. Em vez de listarmos requisitos técnicos, contamos pequenas histórias do ponto de vista do usuário. São as **Histórias de Usuário** (*User Stories*). A analogia perfeita é a de um cozinheiro que, em vez de receber uma lista de ingredientes, recebe um pedido que diz: "Como um cliente com pouco tempo no almoço, eu quero um prato saboroso e rápido, para que eu possa voltar ao trabalho satisfeito e sem atrasos". Essa pequena narrativa contém tudo: o ator (o cliente), a necessidade (o prato) e o motivo (o valor gerado).

Como um

Tipo de usuário

Eu quero

Funcionalidade

Para que

Valor/Objetivo

A estrutura clássica de uma História de Usuário captura essa essência de forma brilhante e simples: **Como um <tipo de usuário>**, eu quero <funcionalidade>, **para que** <eu possa alcançar um objetivo/valor>. Vamos aplicar isso ao nosso item "Cadastro e Login de Usuário". Uma história poderia ser: "Como um *estudante universitário*, eu quero *me cadastrar usando meu e-mail institucional*, para que *o sistema valide automaticamente meu vínculo e eu não precise preencher formulários longos*". Percebe a diferença? Agora a equipe não está apenas construindo uma tela de login; está resolvendo um problema real de um usuário específico, focando na conveniência e na validação automática.

Essa técnica simples, mas profunda, garante que cada linha de código escrita esteja diretamente conectada a uma necessidade real do usuário. Ela promove a empatia e a colaboração, alinhando a equipe de desenvolvimento com os objetivos de negócio. É a base para uma comunicação eficaz, um dos pilares da liderança servidora que as empresas modernas tanto buscam em seus gerentes de projeto.

O Guia para Boas Histórias: O Acrônimo INVEST

Criar boas Histórias de Usuário é uma arte. Uma história mal escrita pode ser tão confusa quanto um requisito técnico vago. Felizmente, a comunidade ágil nos deu um guia prático, um checklist mental para garantir que nossas "receitas" para o desenvolvimento sejam claras e executáveis. Esse guia é o acrônimo **INVEST**.

Pense no INVEST como os critérios de qualidade para cada pedido que chega à cozinha do nosso restaurante. Cada história precisa ser:

Independente

A história deve ser autocontida, sempre que possível. Assim como você pode preparar uma salada sem depender do prato principal estar pronto. Isso dá flexibilidade para priorizar e desenvolver as histórias em qualquer ordem.

Negociável

Uma história não é um contrato rígido. É um convite à conversa entre o *Product Owner* e a equipe. Detalhes podem e devem ser negociados para encontrar a melhor solução.

Valiosa

Se uma história não entrega valor a um usuário ou ao negócio, por que estamos fazendo? Ela precisa ter um "para que..." claro e relevante.

Estimável

A equipe de desenvolvimento precisa conseguir ter uma ideia do tamanho, do esforço necessário. Se a história é muito vaga ("Construir o melhor sistema de busca do mundo"), ela não pode ser estimada.

Pequena (*Small*)

A história deve ser pequena o suficiente para ser concluída dentro de um ciclo de trabalho (uma *Sprint*). Não queremos cozinhar um banquete de casamento inteiro em um único dia, mas sim pratos individuais.

Testável

Precisamos saber o que significa "pronto". Deve haver critérios claros para verificar se a história foi implementada corretamente.

Ao passar cada história do nosso *backlog* pelo filtro do INVEST, garantimos que estamos entregando à equipe de desenvolvimento um trabalho claro, focado em valor e pronto para ser executado. É um mecanismo de controle de qualidade que previne muitos dos problemas comuns em projetos de software.

Priorizando o Caos: Técnicas para Ordenar o Backlog

Nossa lista de desejos, o *Product Backlog*, agora está repleta de Histórias de Usuário bem escritas. O problema? Provavelmente temos dezenas, talvez centenas delas. Se a equipe tentasse fazer tudo de uma vez, seria paralisante. Lembre-se do seu orçamento e tempo limitados para a viagem. Você não pode ir a todos os lugares e fazer tudo na primeira semana. A habilidade mais estratégica de um *Product Owner* é, portanto, a **priorização**.

Imagine que você é o curador de um festival de cinema. Você tem filmes fantásticos (alto valor), mas alguns exigem um equipamento de projeção caríssimo (alto esforço). Outros são curtas-metragens geniais (alto valor) que podem ser exibidos em qualquer sala (baixo esforço). E há aqueles filmes de nicho, que poucas pessoas verão (baixo valor). Como você monta a programação para maximizar o impacto e a satisfação do público com os recursos que tem? Essa curadoria é a essência do refinamento (*grooming*) do *backlog*.

Uma das ferramentas mais eficazes para essa curadoria é a **Matriz de Valor vs. Esforço**. É um gráfico simples de quatro quadrantes. No eixo vertical, temos o "Valor" (para o cliente ou negócio) e, no horizontal, o "Esforço" (complexidade, tempo, custo para desenvolver). Ao plotar cada História de Usuário nessa matriz, o caminho se torna claro:

Alto Valor, Baixo Esforço

Vitórias Rápidas: Comece por aqui! São os curtas-metragens geniais do nosso festival. Em nosso app, talvez seja a opção de "login com Google/Facebook", que é relativamente simples de implementar e agrada muitos usuários.

Alto Valor, Alto Esforço

Épicos/Iniciativas: Estas são as grandes produções. Precisam de planejamento cuidadoso. A "integração com o sistema acadêmico da universidade" se encaixa aqui.

Baixo Valor, Baixo Esforço

Faça se tiver tempo de sobra. Um exemplo seria "permitir que o usuário troque a cor do tema do app".

Baixo Valor, Alto Esforço

Evite a todo custo. São os ralos de recursos.

Essa análise visual, muitas vezes enriquecida por dados de mercado e feedback de usuários — uma forte tendência em gestão orientada por **Data Analytics** —, transforma a priorização de uma arte baseada em intuição para uma decisão estratégica e baseada em evidências.

Quadro Comparativo: Abordagens de Priorização

Após a análise narrativa, um quadro pode ajudar a consolidar as diferenças entre as técnicas comuns de priorização, que muitas vezes são usadas em conjunto.

Técnica	Âmbito/Aplicação	Base/Origem	Exemplo no nosso Projeto
MoSCoW	Categorização rápida de requisitos	Análise de Negócios	Must: Login; Should: Busca com filtros; Could: Gamificação; Won't: Chat entre alunos.
Valor vs. Esforço	Decisão estratégica de portfólio de features	Lean Startup / Análise de Risco	Priorizar "Inscrição com 1 clique" (Alto Valor, Baixo Esforço) antes de "Recomendação com IA" (Alto Valor, Alto Esforço).
First In, First Out (FIFO)	Gestão de bugs ou pequenas melhorias	Filas de produção	Corrigir o bug de digitação reportado ontem antes do que foi reportado hoje.
Cost of Delay	Features sensíveis ao tempo (mercado/receita)	Economia de Desenvolvimento de Produto	Lançar a integração com sistema de pagamento antes de um concorrente (o custo de atrasar é alto).

O Ritual de Partida: A Reunião de Planejamento da Sprint (Sprint Planning)

Com nosso *Product Backlog* priorizado, estamos como um time de montanhismo na base da montanha. O pico (o objetivo do projeto) está visível, e o mapa (o *backlog*) mostra várias rotas possíveis. A *Sprint Planning* é a reunião onde a equipe se junta, olha para o mapa e decide: "Qual será o nosso próximo acampamento-base? E qual trilha exata pegaremos nos próximos dias para chegar lá com segurança?". É o momento de transformar a estratégia de longo prazo em um plano tático de curto prazo.

Este ritual tem um propósito duplo, que se desdobra em duas partes principais. Pense nisso como responder a duas perguntas essenciais antes de iniciar qualquer jornada. A primeira pergunta é "**O quê?**". O *Product Owner* apresenta as Histórias de Usuário do topo do *backlog* e explica o valor de cada uma. A equipe faz perguntas, esclarece dúvidas, e negocia o escopo até que todos tenham um entendimento compartilhado do que precisa ser entregue. É uma conversa, uma negociação para selecionar os itens que cabem na "mochila" da equipe para a próxima etapa da escalada.

Uma vez que o "o quê" está claro, a conversa muda para a segunda pergunta: "**Como?**". Agora, a equipe de desenvolvimento assume o protagonismo. Eles pegam as Histórias de Usuário selecionadas e as quebram em tarefas técnicas menores. Para a história "Cadastro com e-mail institucional", as tarefas poderiam ser: "Criar a interface da tela de cadastro", "Desenvolver o serviço de validação de e-mail", "Configurar o banco de dados" e "Escrever os testes de segurança". É nesse momento que o trabalho se torna tangível e um plano detalhado para os próximos dias começa a se formar.

A reunião culmina na criação de dois artefatos cruciais: o **Sprint Goal**, uma única frase que resume o objetivo da *Sprint* (nosso "chegar ao próximo acampamento"), e o **Sprint Backlog**, a lista de histórias e tarefas que a equipe se comprometeu a realizar.

Estimando o Trabalho: A Arte de Prever o Futuro (Sem uma Bola de Cristal)

"Ok, já sabemos o que vamos fazer e como. Mas... quanto trabalho é tudo isso? Será que conseguimos entregar essas 10 histórias em duas semanas?". Essa é uma das perguntas mais difíceis em qualquer projeto. Estimar em horas é notoriamente impreciso. Uma tarefa que você acha que levará 2 horas pode acabar levando 8 por causa de um problema inesperado. O desenvolvimento de software é complexo e cheio de incertezas. O que fazer?

A solução ágil é genial: em vez de estimar tempo absoluto, estimamos o esforço relativo. Pense em como você descreveria o esforço para ir a lugares diferentes a pé. Ir à padaria na esquina é "fácil". Ir ao shopping no centro é "médio". Subir uma montanha é "muito difícil". Você não disse quantos minutos levaria, mas deu uma medida relativa de esforço, complexidade e risco. É isso que são os **Story Points**. São unidades sem dimensão que a equipe usa para comparar o tamanho das histórias. Uma história de 1 ponto é trivial. Uma de 8 pontos é significativamente mais complexa.

📄 Para chegar a um consenso sobre esses pontos, as equipes frequentemente usam uma técnica chamada **Planning Poker**. Não, não envolve apostas, mas sim colaboração.

Para cada história, cada membro da equipe escolhe uma carta com um número (geralmente da sequência de Fibonacci: 1, 2, 3, 5, 8, 13...) que representa sua estimativa de esforço. Todos revelam suas cartas ao mesmo tempo. Se os números forem parecidos, ótimo! Se forem muito diferentes (um votou 2 e outro 8), isso gera uma conversa valiosa. O desenvolvedor que votou 8 pode ter visto um risco que o outro não viu. Essa técnica utiliza a inteligência coletiva para chegar a estimativas mais realistas e expor problemas ocultos antes mesmo de o trabalho começar.

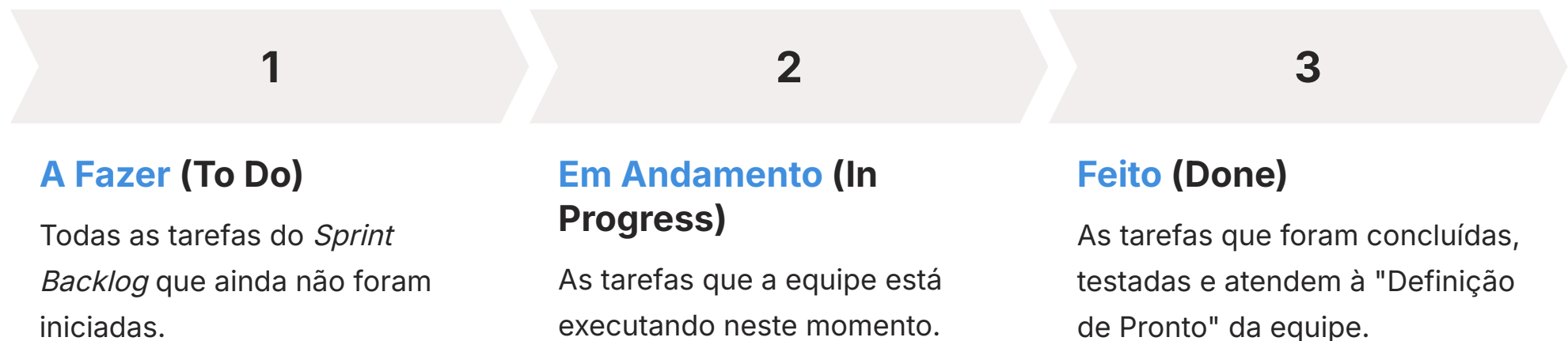
Essa abordagem se conecta a uma tendência moderna: o uso de **Inteligência Artificial (IA)** em ferramentas de gestão. Plataformas como o Jira estão começando a usar IA para analisar dados de *sprints* passadas e sugerir estimativas, ajudando as equipes a refinar ainda mais suas previsões e a tomar decisões baseadas em seu próprio histórico de desempenho.

Montando o Quebra-Cabeça: O Sprint Backlog e a Transparência Visual

O plano está traçado, o trabalho estimado, e a equipe está comprometida com um objetivo. Mas como garantir que todos permaneçam na mesma página durante as próximas semanas? Como o gerente de projetos (ou *Scrum Master*) pode saber o andamento sem precisar perguntar a cada pessoa a cada hora? A resposta está em tornar o trabalho visível.

O **Sprint Backlog** é o resultado final da *Sprint Planning*. É a lista de Histórias de Usuário que a equipe se comprometeu a entregar, junto com as tarefas detalhadas necessárias para completá-las. Se o *Product Backlog* é o mapa completo da jornada, o *Sprint Backlog* é a folha de rota detalhada para o trecho atual. É um plano feito *pela* equipe, *para* a equipe, o que gera um enorme senso de propriedade e comprometimento.

A forma mais comum de gerenciar o *Sprint Backlog* é através de um quadro visual, seja ele um quadro branco com post-its ou uma ferramenta digital como Trello, Asana ou Jira. Esse quadro, muitas vezes chamado de **Kanban Board**, é simples e poderoso. Ele possui, no mínimo, três colunas:



Ao longo da *Sprint*, os membros da equipe movem os cartões (tarefas) da esquerda para a direita. Esse quadro se torna o ponto central de informação do projeto. Qualquer pessoa, a qualquer momento, pode olhar para o quadro e entender exatamente o progresso, identificar gargalos (se muitas tarefas se acumulam em "Em Andamento") e celebrar as vitórias à medida que a coluna "Feito" vai se enchendo. É a materialização da transparência, um valor fundamental da agilidade.

O Foco na Liderança e na Sustentabilidade do Ritmo

Pode parecer que todo esse processo de planejamento é apenas sobre tarefas, quadros e pontos. Mas a verdadeira mágica acontece na camada humana. O papel do gerente de projetos aqui se transforma. Ele não é mais um "chefe" que distribui tarefas e cobra prazos. No mundo ágil, especialmente em papéis como o de *Scrum Master*, ele se torna um **líder servidor e facilitador**. Seu trabalho é garantir que a equipe tenha tudo o que precisa para ter sucesso, remover obstáculos e proteger o time de interrupções externas.

Essa mudança de papel exige um foco intenso em **habilidades de liderança e inteligência emocional**. A *Sprint Planning* não é apenas uma reunião técnica; é uma negociação, uma dinâmica de equipe que exige comunicação eficaz, escuta ativa e a habilidade de construir consenso. O líder aqui não dá as respostas, mas faz as perguntas certas para que a equipe encontre as melhores soluções. Ele cria um ambiente de segurança psicológica onde todos se sentem à vontade para dar estimativas honestas sem medo de punição.



Proteção da Equipe

O líder servidor protege a equipe de interrupções externas e mudanças constantes de escopo durante a Sprint.



Inteligência Emocional

Facilita conversas difíceis e cria um ambiente de segurança psicológica para estimativas honestas.



Sustentabilidade

Promove um ritmo sustentável de trabalho, combatendo o burnout e as horas extras excessivas.

Além disso, há uma conexão importante com o conceito de **Sustentabilidade e ESG** (*Environmental, Social, and Governance*). O planejamento ágil, ao focar em *sprints* de curta duração com um volume de trabalho estimado pela própria equipe, promove um ritmo sustentável. Ele combate a cultura de horas extras excessivas e o *burnout*, que são problemas sociais graves na indústria de TI. Ao se comprometer com um volume de trabalho realista para uma *sprint*, a equipe e a organização estão, na prática, gerenciando seus recursos humanos de forma mais responsável e sustentável, o que é um pilar da governança social (o "S" do ESG). Um time saudável e motivado é um time mais produtivo e inovador a longo prazo.

Conectando as Peças: Da Visão ao Plano de Ação

Vamos recapitular a jornada que fizemos nesta aula. Partimos de uma visão ampla, quase um sonho, para o nosso projeto de aplicativo. Esse sonho foi traduzido em uma lista de desejos organizada e priorizada, o **Product Backlog**. Para tornar esses desejos compreensíveis e focados no cliente, nós os reescrevemos como **Histórias de Usuário**, garantindo que cada pedaço de trabalho estivesse atrelado a um valor real.

Com as histórias mais importantes no topo, realizamos o ritual da **Sprint Planning**. Nesse momento de colaboração intensa, a equipe selecionou o que poderia entregar nas próximas semanas, definindo um **Sprint Goal** claro. Em seguida, quebraram essas histórias em tarefas técnicas e estimaram o esforço relativo usando **Story Points**. O resultado final desse processo é o **Sprint Backlog**, um plano de ação detalhado e visível para todos, geralmente gerenciado em um quadro Kanban.

01	02	03
Visão do Projeto	Product Backlog	Histórias de Usuário
Ideia ampla transformada em lista organizada	Lista priorizada de funcionalidades	Funcionalidades focadas no valor do cliente
04	05	
Sprint Planning	Sprint Backlog	
Seleção e detalhamento do trabalho	Plano de ação executável	

O que acabamos de construir não é apenas um plano; é um acordo. Um acordo entre o negócio (representado pelo *Product Owner*) e a equipe de desenvolvimento. É um compromisso baseado em transparência, colaboração e realismo. Deixamos para trás a incerteza de um escopo gigante e abraçamos a clareza de um objetivo de curto prazo. Essa é a essência do planejamento ágil: criar foco, reduzir o desperdício e começar a entregar valor o mais rápido possível.

Mas, como sabemos, nenhum plano sobrevive ao contato com a realidade. O que acontece quando o trabalho começa? Como acompanhamos o progresso dia a dia e lidamos com os imprevistos? Essa é a ponte para a nossa próxima aventura.

Consolidação e Próximos Passos

Nesta aula, transformamos a abstração de uma ideia de projeto em um plano de trabalho concreto e ágil. Vimos como o **Product Backlog** serve como uma fonte única da verdade e como as **Histórias de Usuário** garantem o foco no cliente. Dominamos o processo da **Sprint Planning**, onde a equipe define "o quê" será feito e "como", resultando em um **Sprint Backlog** claro e um objetivo motivador. Esse processo iterativo é o motor que impulsiona a entrega de valor contínua nos projetos modernos de TI.

Em Prática

- Antes de iniciar qualquer nova funcionalidade, formule-a como uma História de Usuário: "Como um..., eu quero..., para que...".
- Use uma matriz de Valor vs. Esforço como uma ferramenta rápida para desempates na hora de priorizar tarefas.
- Trate o planejamento da *Sprint* como uma conversa colaborativa, não uma reunião de atribuição de tarefas de cima para baixo.
- Torne o trabalho da equipe visível para todos através de um quadro físico ou digital. A transparência gera confiança.

Autoavaliação

1. **(Nível Fácil)** Durante a reunião de Planejamento da Sprint (Sprint Planning), qual é o principal papel do Product Owner na primeira parte da reunião ("O Quê?")?
 - a) Definir as tarefas técnicas para cada item.
 - b) Estimar o esforço de cada História de Usuário em Story Points.
 - c) Apresentar e esclarecer os itens de maior prioridade do Product Backlog.
 - d) Criar o quadro Kanban para a Sprint.
2. **(Nível Médio)** Uma equipe de desenvolvimento está analisando a seguinte História de Usuário: "Como usuário, eu quero um sistema melhor". Qual critério do acrônimo INVEST esta história mais claramente viola?
 - a) Independente e Negociável.
 - b) Valiosa e Pequena (Small).
 - c) Estimável e Testável.
 - d) Negociável e Valiosa.
3. **(Nível Difícil - Estilo Concurso)** De acordo com as práticas do Scrum, o Sprint Backlog é criado durante a Sprint Planning e consiste em dois elementos fundamentais. Assinale a alternativa que descreve corretamente esses elementos.
 - a) O Sprint Goal e a lista de impedimentos previstos.
 - b) As Histórias de Usuário selecionadas para a Sprint e o plano para entregá-las (tarefas).
 - c) A Definition of Done e as métricas de performance (KPIs) da equipe.
 - d) O Product Backlog completo e a capacidade da equipe para a Sprint.
4. **(Nível Especialista)** A técnica de estimativa que utiliza o consenso do grupo para atribuir um valor relativo de esforço, complexidade e incerteza a um item do backlog é conhecida como:
 - a) Análise de Ponto de Função.
 - b) Planning Poker.
 - c) Método MoSCoW.
 - d) Gráfico de Burndown.

Questão Discursiva Curta:

Descreva com suas palavras por que um Product Backlog é considerado um "artefato vivo" e como essa característica o diferencia de um escopo de projeto definido em uma abordagem tradicional (cascata).

Gabarito: 1-C, 2-C, 3-B, 4-B

- ❏ **Resposta Sugerida para a Discursiva:** O Product Backlog é um "artefato vivo" porque está em constante evolução; novos itens podem ser adicionados, existentes podem ser removidos, e a priorização é reavaliada continuamente para maximizar o valor. Isso o diferencia do escopo tradicional (cascata), que é fixo e detalhado no início do projeto, sendo resistente a mudanças.

Próxima Aula

Agora que temos nosso plano para a primeira *Sprint*, o que acontece a seguir? Na **Aula 54 – Projeto Final – Parte 4: Simulação de Execução e Controle**, vamos colocar a mão na massa. Simularemos a execução dessa *Sprint*, aprenderemos a acompanhar o progresso com ferramentas visuais como o Gráfico de Burndown e a conduzir as cerimônias diárias (*Daily Scrums*) para manter a equipe sincronizada e produtiva.

Recursos Adicionais

- **Artigo:** *User Stories with Examples and a Template* (Mike Cohn) - Uma referência clássica do cofundador da Scrum Alliance sobre como escrever histórias eficazes.
- **Vídeo:** *Scrum in 10 Minutes* (Axosoft) - Uma animação curta e excelente que revisa todo o fluxo do Scrum, incluindo a Sprint Planning.

NOTA IMPORTANTE: As informações e práticas de gerenciamento de projetos desta aula estão atualizadas até 2025. Consulte sempre guias oficiais como o Scrum Guide e o PMBOK Guide para verificar as versões mais recentes das metodologias.